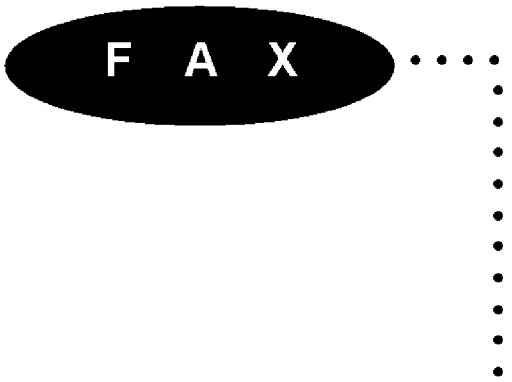

10733 Calston Way
San Diego, CA 92126

To: Paul Kang, Patent Examiner

Fax number: 1 571 273-3882

From: Haw-minn Lu

Fax number:

Business phone: 8583827513

Home phone: 8583827513

Date & Time: 4/9/2009 2:08:54 AM

Pages: 11

Re: Proposed Amendment

Attached is a proposed amendment per our conversation Tuesday

Per our discussion on April 7, 2009, regarding Application # 10/761,894, Applicant suggests the following amendments as shown on the listing of claims beginning on the following page.

Applicant would also suggest the following amendment to the specification:

[074] If the reply is a RSET, SEND, ~~SCML~~SOML, SAML, VRFY, NOOP, EXPN, HELP, or TURN command as shown in **Fig. 8E** then the algorithm relays the reply to MTA_1 and waits for a new reply.

This was a typographical error. There is no SMTP command called "SCML" but rather SOML (see *Postel* Reference §3.4 on p. 11). An amendment is also suggested to claim 17. In the event such an amendment is not appropriate in an Examiner's amendment, Applicant is willing to submit it as a §1.312 amendment.

Regarding the amendments to the claims, per our discussion an amendment was made to claim 1 and 16 to address potential §101 concerns. Applicant proposes amendments to claims 9-11 to correct consistency of language missed in the previous amendment of 9/29/2008. In addition to deleting system to address potential §101 concerns and correcting the SCML/SOML typographical error in claim 17. Applicant proposes an amendment to step (ll) to make it consistent with paragraph [059]. There is no 343 reply it is a typographical error and should be a 354 reply. Rather than amend claim 18, Applicant would suggest representing the claim as claim 20 which essentially is a method claim incorporating all limitations of claim 1, which the Examiner found is allowable. Some terms in the preamble used as an antecedent basis in claim 1 for its dependents claims have been omitted since claim 18/20 has no dependent claims.

Should there be any discrepancies or remaining issues you may contact me on my mobile phone 858-382-7513.

Best Regards,
/Haw-minn Lu/

Haw-minn Lu, Reg No. 55,407

Proposed Listing of Claims

1) (Currently Amended) ~~An~~ A networked computer comprising an unsolicited message rejecting communications processor connected between message transfer agents

MTA_0 with an Internet address IP_0, a from-address A_0, a declared domain D_0, and a real domain DD_0, and

MTA_1 with an Internet address IP_1, a domain D_1, and a to-address A _1 comprising:

a) monitoring means for monitoring the communications between MTA_0 and MTA_1;

b) determining means for determining if the communications contains a message that is unsolicited;

c) intercepting means for intercepting a .\r\n end-of-message indicator reply from MTA_0, forcing MTA_0 to QUIT its connection with the unsolicited message rejecting communications processor by sending an error reply to MTA_0 if the message is determined to be unsolicited;

wherein the unsolicited message rejecting communications processor does not intercept communications between MTA_0 and MTA_1 before a .\r\n end-of-message indicator reply from MTA_0 is received by the unsolicited message rejecting communications processor.

2) (Previously Presented) The unsolicited message rejecting communications processor in Claim 1, further includes an allow_address database and wherein the determining means determines if the message is not unsolicited by checking if the IP_0 is in the allow_address database.

3) (Previously Presented) The unsolicited message rejecting communications processor in Claim 1, further includes a prevent_address database and wherein the determining means determines if the message is unsolicited by checking if IP_0 is in the prevent_address database.

4) (Previously Presented) The unsolicited message rejecting communications processor in Claim 1, further includes access to an open relay database and wherein the determining means determines if the message is unsolicited by checking if IP_0 is in the open relay database.

5) (Previously Presented) The unsolicited message rejecting communications processor in Claim 1, further includes access to a DNS (domain name server) database and wherein the determining means determines if the message is unsolicited by checking if IP_0 has a domain name entry DD_0 in the DNS database.

6) (Previously Presented) The unsolicited message rejecting communications processor in Claim 1, further includes a bad_from database and wherein the determining means determines if the message is unsolicited by checking if the from-address A_0 is in the bad_from database.

7) (Previously Presented) The unsolicited message rejecting communications processor in Claim 1, further includes a suspect_domain database and wherein the determining means determines if the message is unsolicited by checking if the real domain DD_0 matches the domain of the from-address A_0 and the domain of the from-address A_0 is in the suspect_domain database.

8) (Previously Presented) The unsolicited message rejecting communications processor in Claim 1, wherein the determining means determines if the message is unsolicited by checking if the from-address A_0 matches the to-address (A_1).

9) (Currently Amended) The unsolicited message rejecting communications processor in Claim 1, further includes a no_filter database and wherein the determining means

determines if the message is to be ~~blocked~~ rejected if it is determined to be unsolicited.

10) (Currently Amended) The unsolicited message rejecting communications processor in Claim 1, wherein the determining means determines if ~~a~~ the message is unsolicited by checking if the declared domain D_0 is the same as the domain D_1.

11) (Currently Amended) The unsolicited message rejecting communications processor in Claim 1, wherein the determining means determines if ~~a~~ the message is unsolicited by checking if the declared domain D_0 does not match the real domain DD_0 and the declared domain D_0 is in the suspect_domain database.

12) (Previously Presented) The unsolicited message rejecting communications processor in Claim 1, further includes a bad_word database and wherein the determining means determines if the message is unsolicited by checking if the subject line of the message contains any words in the bad_word database.

13) (Previously Presented) The unsolicited message rejecting communications processor in Claim 1, further includes a bad_fingerprint database and wherein the determining means determines if the hash "fingerprint" of a portion of the body of the message is in the bad_fingerprint database.

14) (Previously Presented) The unsolicited message rejecting communications processor in Claim 1, further includes a rejected_connection database which logs the time, from-address A_0, to-address A_1, and the reason for the rejection if the message is determined to be unsolicited.

15) (Previously Presented) The unsolicited message rejecting communications processor in Claim 1, further includes an allowed_connection database which logs the time and to-address A_1 if the message is determined not to be unsolicited.

16) (Currently Amended) A method for

a receiving networked computer ~~system~~ with an Internet connection, a message transfer agent MTA_1, an Internet address IP_1, a to-address A_1, and an operating system capable of executing the method

to reject unsolicited messages from

a transmitting networked computer system with an Internet connection and a message transfer agent MTA_0, an Internet address IP_0, a from-address A_0, a declared domain D_0, and from a real domain DD_0

comprising the steps of:

a) waiting for a new SMTP connection request;
b) relaying and monitoring the replies from MTA_0 to MTA_1;
c) relaying replies from MTA_1 to MTA_0;
d) intercepting the .\r\n end-of-message indicator reply from MTA_0 to MTA_1;
e) determining if the message is unsolicited by analyzing the monitored replies;
f) releasing the intercepted .\r\n end-of-message reply if the message is determined not to be unsolicited; and
g) sending an error reply to MTA_0 to force MTA_0 and MTA_1 to close down their connection;

whereby MTA_1 controls the interaction between MTA_0 and MTA_1 until a .\r\n end-of-message indicator reply is received from MTA_0.

17) (Currently Amended) A method for

a receiving networked computer ~~system~~ with an Internet connection, a DNS server, and an open relay database, a message transfer agent MTA_1, an IP address IP_1, a domain name D_1, a to-address A_1, an allow_address database, a prevent_address database, a suspect_domain database, a bad_from database, a no_filter database, a yes_filter database, a bad_word database, a bad_fingerprint, a rejected_connection database, an allowed_connection database, and an operating system capable of executing the method

to reject unsolicited messages from

a transmitting networked computer system with an Internet connection, a message transfer agent MTA_0, an IP address IP_0, a declared domain D_0, a real domain DD_0, and a from-address A_0

comprising the steps of:

a) waiting for a SMTP connection request on the receiving networked computer system's Internet connection;
b) sending a 220 reply to MTA_0 to acknowledge the SMTP connection request;
c) extracting the IP address IP_0 from the SMTP connection request;
d) requesting the domain name DD_0 for IP_0 from the DNS server;
e) testing if the domain name DD_0 is "no name";
f) testing if IP_0 is in the open relay database;
g) testing if IP_0 is in the allow_address database;
h) testing if IP_0 is in the prevent_address database[[,]];
i) requesting a connection with MTA_1;
j) waiting for a 220 reply from MTA_1 to acknowledge the requested connection;
k) waiting for a reply from either MTA_0 or MTA_1;
l) jumping to step o) if the reply is not from MTA_1;
m) relaying the reply from MTA_1 to MTA_0;

n) jumping to step k) to wait for a new reply;

o) jumping to step u) if the reply from MTA_0 is not a **HELO**;

p) extracting the declared domain D_0 from the reply;

q) testing if the declared domain D_0 matches the domain D_1;

r) testing if the declared domain D_0 does not match the real domain DD_0 AND the declared domain D_0 is in the suspect_domain database;

s) relaying the HELO reply from MTA_0 to MTA_1;

t) jumping to step k) to wait for a new reply;

u) jumping to step aa) if reply from MTA_0 is not a **MAIL**;

v) extracting the from-address A_0;

w) testing if A_0 is in the bad_from database;

x) testing if DD_0 does not match the domain of A_0 and the domain of A_0 is in the suspect_domain database;

y) relaying MAIL reply to MTA_1;

z) jumping to step k) to wait for a new reply;

aa) jumping to step ii) if the reply from MTA_0 is not a **RCPT**;

bb) extracting the to-address A_1;

cc) testing if A_1 is in the no_filter database;

dd) testing if A_0 matches A_1;

ee) testing if A_0 is in the no_filter database;

ff) testing if A_0 is in the yes_filter database;

gg) relaying RCPT reply to MTA_1;

hh) jumping to step k) to wait for a new reply;

ii) jumping to step yy) if the reply from MTA_0 is not **DATA**;

jj) relaying DATA to MTA_1;

kk) waiting for a 354 reply from MTA_1;

ll) relaying ~~a 343~~ the 354 reply to MTA_0;

mm) wait for the body of the message;

nn) relaying the body of the message to MTA_1;

oo) waiting for a .\r\n end-of-message indicator;

pp) testing if any word in the subject line of the message is in the bad_word database;

qq) testing if the hash "fingerprint" of a portion of the message is in the bad_fingerprint database;

rr) jumping to step vv) if NOT(t_allow OR t_no_filter OR OR NOT t_yes_filter OR NOT (t_prevent OR t_open OR t_DD-) OR t_bad_from OR t_suspect_domain OR t_echo_domain OR t_forged_domain OR t_bad_word OR t_bad_fingerpring)) ;

ss) logging the time and the to-address A_1 in the allowed_connection database;

tt) relaying the .\r\n end-of-message indicator reply to MTA_1 to continue the conversation;

uu) jumping to step k) to wait for a new reply;

vv) logging the time, the from-address A_0, the to-address A_1, and the reason for rejecting the connection in the rejected_connection database;

ww) sending a 554 reply to MTA_0 to terminate the conversation;

xx) jumping to step k) to wait for a new reply;

yy) jumping to step ggg) if the reply from MTA_0 is not **RSET, SEND, ~~SCML~~SOML, SAML, VRFY, NOOP, EXPN, HELP,** or **TURN;**

zz) relaying the reply to MTA_1;

aaa) jumping to step j) to wait for a new reply;

bbb) jumping to step ddd) if the reply from MTA_0 is not a **QUIT**;

ccc) relaying the QUIT reply to MTA_1;

ddd) waiting for a 221 reply from MTA_1

eee) relaying the ~~a~~ 221 reply from MTA_1 to MTA_0;

fff) jumping to step a) to wait for a new connection;

ggg) sending a 500 reply to MTA_0 to signal a syntax error; and

hhh) jumping to step a) to wait for a new connection.

18) (Cancelled)

19) (Cancelled)

20) (New) A method in a networked computer comprising an unsolicited message rejecting communications processor connected between message transfer agents MTA_0 and MTA_1, said method comprising:

monitoring communications between MTA_0 and MTA_1;

determining if the communications contains a message that is unsolicited; and

intercepting a .\r\n end-of-message indicator reply from MTA_0, forcing MTA_0 to QUIT its connection with the unsolicited message rejecting communications processor by

sending an error reply to MTA_0 if the message is determined to be unsolicited;

wherein the unsolicited message rejecting communications processor does not intercept communications between MTA_0 and MTA_1 before a .\r\n end-of-message indicator reply from MTA_0 is received by the unsolicited message rejecting communications processor.